6A

УДК 330.43

Е.А. Арсирий, С.Г. Антощук, О.А. Игнатенко, Б.Ф. Трофимов
Одесский национальный политехнический университет
Одесса, просп. Т. Шевченко, 1

Автоматизация разработки и обновления семантического ядра сайта с динамическим контентом

E.A. Arsirij, S.G. Antotshuk, O.A. Ignatenko, B.F. Trofimov
Odessa National Polytechnic University
Odessa, T. Shevchenko Avenue

Automation of Development and Updating for Semantic Kernel of a Site with Dynamic Content

О.О. Арсірій, С.Г. Антощук, О.О. Ігнатенко, Б.Ф. Трофимов
Одеський національний політехнічний університет
Одеса, просп. Т. Шевченка, 1

Автоматизація розробки і оновлення семантичного ядра сайта з динамічним контентом

В результате анализа проблем продвижения в поисковых системах веб-ресурсов с динамическим контентом предложена методика автоматизации разработки и обновления семантического ядра сайта на основе создания ассоциативных правил с помощью алгоритма поиска популярных наборов Apriori в базе поисковых транзакций. Применение методики позволило повысить полноту и точность, а также снизить время разработки и обновления семантического ядра сайта типа интернет-витрины и магазина.
Ключевые слова: семантическое ядро сайта, динамический контент, поисковая система

As the result of analysis of the problems of promotion for web-resources with dynamic content in the search systems, the techniques for automation and updating of semantic kernel of a site are given on the basis of creation of content rules with the help of algorithm for search of popular sets Apriori in the transaction search base. The use of methods allows improving accuracy and reducing time for development and updating of semantic kernel of such site as online shopping.
Key Words: semantic kernel of a site, dynamic content, search engine.

У результаті аналізу проблем просування в пошукових системах веб-ресурсів з динамічним контентом запропонована методика автоматизації розробки та оновлення семантичного ядра сайту на основі створення асоціативних правил за допомогою алгоритму пошуку популярних наборів Apriori в базі пошукових транзакцій. Застосування цієї методики дозволило підвищити повноту і точність, а також зменшити час розробки та оновлення семантичного ядра сайта типу інтернет-вітрини та магазину.
Ключові слова: семантичне ядро сайта, динамічний контент, пошукова система.

Введение

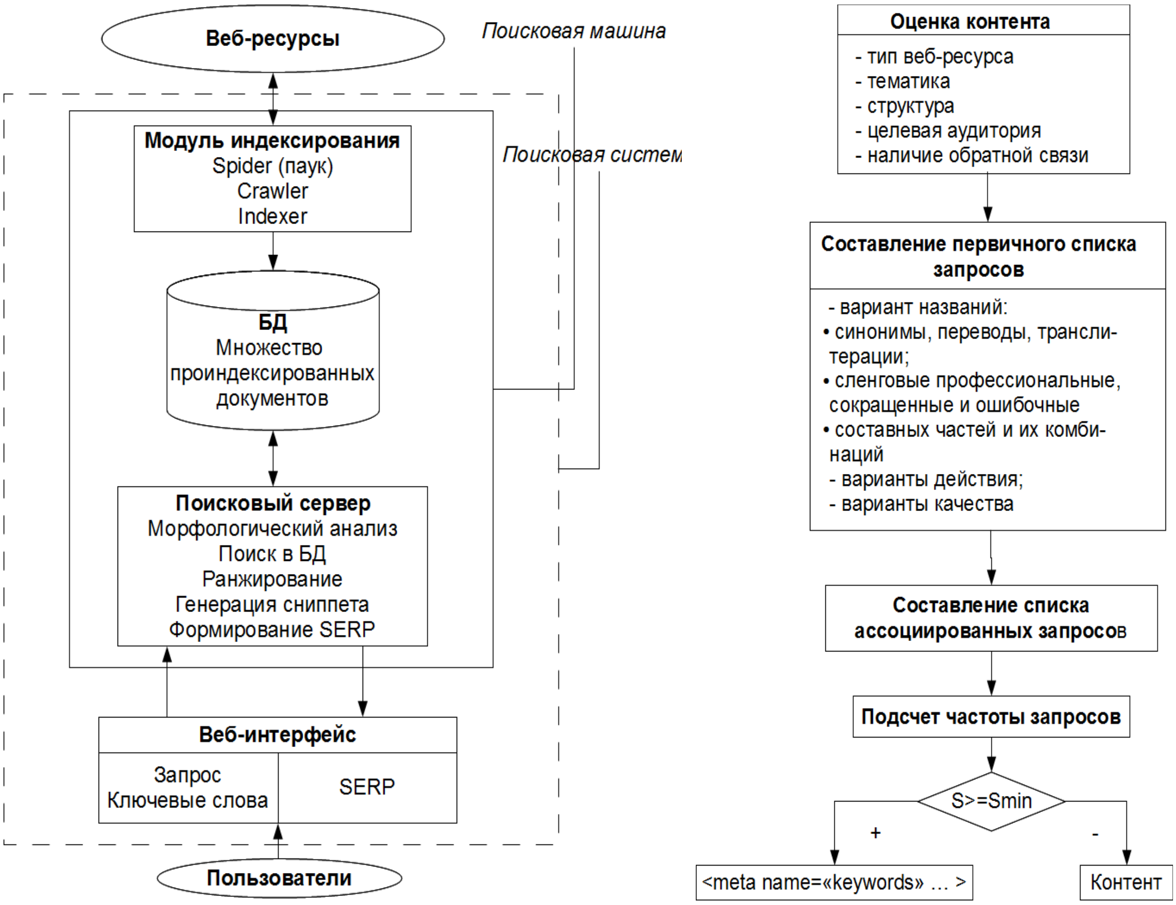
В век информационных технологий успех практически любого бизнеса в достаточно большой степени зависит от способов виртуального представления фирмы в сети Интернет. При этом целью разработки контента веб-ресурса (сайта) фирмы является

предоставление информации, которая была бы способна заставить пользователя думать и вести себя в направлении, выгодном реальному бизнесу. С другой стороны, известно, что доля «поискового трафика» любого сайта (число посетителей, пришедших от поисковых выдач от общей посещаемости сайта) является преобладающей [1], [2]. Поэтому при разработке контента сайта большое внимание уделяется SEO (search engine optimization) – комплексу мер, направленных на продвижение веб-ресурса к верхним позициям поисковой системы (ПС) с целью увеличения его посещаемости. Известно, что одним из ключевых этапов SEO является разработка семантического ядра сайта (СЯС), которая, как правило, выполняется специалистами вручную и требует больших временных затрат [4]. Такое положение является особенно недопустимым при разработке и обновлении СЯС с динамическим контентом, когда SEO-специалисты не успевают вовремя реагировать на изменяющиеся предпочтения и действия пользователей, внешнее интернет-окружение сайта, а также его содержимое. Поэтому актуальным является создание методики разработки и обновления СЯС, применение которой SEO-специалистами позволило сократить время на достижение и поддержание лидирующих позиций сайта в поисковых выдачах. Для создания методики авторам необходимо было: проанализировать связь между этапами и процедурами работы ПС и разработки или обновления СЯС и предложить способ ее описания; определить требования к формированию транзакционной базы данных в терминах анализа связей и разработать базу поисковых транзакций; разработать методику применения анализа связей к базе поисковых транзакций; предложить методику реализации поиска популярных наборов с помощью алгоритма Apriori и создания ассоциативных правил на основе найденных популярных наборов для разработки или обновления СЯС.

Анализ этапов и процедур работы ПС и разработки СЯС

ПС представляет собой сайт, состоящий из веб-интерфейса для пользователя и поисковой машины, которая является движком, обеспечивающим функциональность ПС. Поисковая машина состоит из модуля индексирования, базы данных (БД) проиндексированных документов и поискового сервера, занимающегося анализом и обработкой запросов пользователей. Модуль индексирования состоит из трех вспомогательных программ (роботов) – spider (паук), crawler (путешествующий паук) и indexer (индексатор). Паук скачивает веб-документы с помощью протокола HTTP, извлекает ссылки и перенаправления и сохраняет текст в следующем формате: URL, дата скачивания, http-заголовок ответа сервера, тело страницы (html-код). Crawler обрабатывает найденные пауком ссылки и осуществляет дальнейшее направление паука. Indexer разбирает html-код страницы на составные части, такие как заголовки (title), подзаголовки (subtitles), метатэги (meta tags), текст, ссылки, структурные и стилевые особенности и т.д, анализирует их на основе различных лексических и морфологических алгоритмов с целью последующего ранжирования по степени важности. При этом найденным словам и словосочетаниям присваиваются весовые коэффициенты в зависимости от того, сколько раз и где они встречаются (в заголовке страницы, в начале или в конце страницы, в ссылке, в метатэге и т.п.). В результате формируется файл, содержащий индекс, который может быть довольно большим. Для уменьшения его размеров прибегают к минимизации объема информации и сжатию файла, а также решают задачи определения дубликатов и «почти дубликатов».

Результаты индексирования записываются в базу данных (БД) проиндексированных документов (рис. 1 а).



а) б)

Рисунок 1 – Обобщенная схема этапов и процедур:
а) – работы ПС; б) – разработки СЯС

Поисковый сервер является важнейшим элементом всей ПС, так как от алгоритмов, которые лежат в основе его функционирования, зависит качество и скорость поиска. Принцип его работы заключается в следующем. Полученный от пользователя запрос (ключевые слова) подвергается морфологическому анализу для получения информационного окружения. При этом выделяются информационные (поиск сведений), транзакционные (совершение действия), нечеткие (общие) и навигационные (прямой адрес) запросы. Поиск документов по их содержанию называется семантическими. Информационное окружение передается специальному модулю ранжирования, задача которого состоит в поиске *html* страниц в БД проиндексированных документов, сортировке и выдаче в порядке релевантности. При этом для оценки релевантности найденных документов, как правило, используют TF-IDF-меру, согласно которой релевантность документа будет выше, если слово или словосочетание из запроса *чаще* встречается в найденном документе (частота слова – *term frequency* – TF) и *реже* других документах БД – IDF (обратная частота документа – *inverse document frequency* – IDF). Если необходимо, прядок выдачи документов может быть изменен пользователем путем задания

дополнительных условий (расширенный поиск). Далее генерируется сниппет, то есть для каждого найденного документа из таблицы документов извлекаются заголовок, краткая аннотация, наиболее соответствующая запросу и ссылка на сам документ, причем найденные слова подсвечиваются. Полученные результаты поиска передаются пользователю в виде SERP (*Search Engine Result Page*) – страницы выдачи поисковых результатов. Таким образом, основой работы всех ПС является определение так называемых «ключевых слов» веб-ресурса. Из списка таких слов состоит семантическое ядро сайта (СЯС). СЯС представляет собой список ключевых слов и их комбинаций, записанных в метатэги keywords и распределенных в контенте сайта, а именно, в тэге *title*, в *alt*-атрибутах, в ссылочном тексте внутренних и внешних ссылок, в выделениях жирным и наклонным шрифтом, в начале контента сайта, в названии файлов, в URL и др. При этом от полноты и точности разработки СЯС зависит положение сайта в списке выдач ПС.

Разработка СЯС является ключевым этапом SEO и состоит из ряда интеллектуальных, трудноформализуемых этапов и процедур, для реализации которых необходимы большие временные и человеческие ресурсы (рис. 1 б).

На первом этапе необходимо оценить контент сайта, определив его тип (интернет-магазин, новостной блог, сайт-визитка и пр.), тематику, структуру, целевую аудиторию и необходимость обратной связи с пользователями. Следующим этапом будет создание первичного списка запросов. Для этого можно использовать различные варианты названий товаров, услуг, самого сайта, различные действия, предоставляемые пользователям, и варианты качества товара или услуг [1]. Затем составляется список ассоциированных запросов с помощью средств статистики поисковых систем (wordstat.yandex, adstat.rambler, google/insights/search/, adwords.google и др.) и подсчитывается частота ключевых слов. Ключевые слова с наибольшей частотой помещают в метатэги keywords, с меньшей – распределяют по контенту сайта. Однако, для сайтов с динамическим контентом, таких как интернет-витрина, магазин, новостной блог, где меняется ассортимент товаров, их популярность, новости, заголовки и пр., перечисленные этапы разработки СЯС необходимо повторять достаточно часто. При этом длительность выполнения каждого этапа может значительно задерживать необходимую периодичность повторения, что приводит к снижению полноты и точности СЯС, а сайт теряет свои позиции в SERP. Для сокращения времени разработки и обновления СЯС с динамическим контентом без потери полноты и точности в данном исследовании предлагается использовать анализ связей (link analysis), позволяющий сгенерировать правила количественного описания взаимной связи между двумя и более ключевыми словами, объединенными в одном семантическом запросе. Такие правила в терминах анализа связей называются *ассоциативными*, а запрос представляет собой некоторое множество событий, происходящих совместно, и образует *транзакцию*.

Методика анализа связей в базе поисковых транзакций

Транзакционная или операционная БД (Transaction database) представляет собой двумерную таблицу, которая состоит из номера транзакции (TID) и перечня ключевых слов, составивших запрос во время этой транзакции. Пример фрагмента базы поисковых транзакций (БПТ) для интернет витрины Konica-Digital показан в табл. 1, где TID – уникальный идентификатор, определяющий каждую поисковую

транзакцию. На основе имеющейся БПТ необходимо найти связи между событиями, которые представляют собой поисковые запросы пользователей.

Таблица 1 – База поисковых транзакций

TID	*Поисковые запросы*				*TID*	*Поисковые запросы*		
1	флешки,	онлайн			8	фото,	рамки,	онлайн
2	*фото,*	*альбом*			9	печать,	фото	
3	*фото*,	*альбом,*	онлайн		10	интернет,	магазин,	фото
4	фото,	рамки			11	*фото*,	магазин,	*альбом*
5	фото,	рамки,	купить,	онлайн	12	фото,	магазин,	рамки
6	печать,	фото,	онлайн		13	рамки,	*альбом,*	*фото*
7	фото,	магазин			14	печать,	флешки	

В терминах анализа связей пусть $I = \{i_1, i_2, i_3, \ldots i_n\}$ – множество (набор) ключевых слов, называемых элементами. Пусть M – множество транзакций из БПТ, где каждая транзакция T с уникальным номером *TID* – это набор элементов из I, $T \subseteq I$. При этом, длиной транзакции T с номером *TID* считается количество элементов из I, входящих в T – D_{TID}. Говорят, что транзакция T содержит A, некоторый набор элементов из I, если $A \subseteq T$. Ассоциативным правилом (АП) состоящим из двух наборов элементов называется импликация $A \rightarrow B$, *где* $A \subset I$, $B \subset I$ *и* $A \cap B = \varnothing$. При этом А называют условием (antecedent), а В – следствием (consequent) и говорят «Если A то B». Можно выделить объективные (независимые от конкретного приложения) меры значимости АП. К таким мерам, описывающим связь между наборами элементов, которые соответствуют условию и следствию, относят поддержку – *supp* (support) и достоверность *conf* – (confidence) [3].

Говорят, что правило $A \rightarrow B$ имеет *поддержку supp*, если *supp%* транзакций из M, содержат $A \cup B$ (условие и следствие)

$$supp(A \rightarrow B) = supp(A \cup B) \tag{1}$$

Достоверность conf правила – отношение количества транзакций, содержащих условие A и следствие B, к количеству транзакций, содержащих только условие A – показывает какова вероятность того, что из A следует B.

$$conf(A \rightarrow B) = supp(A \cup B)/supp(A) \tag{2}$$

При этом говорят, правило $A \rightarrow B$ справедливо с достоверностью *conf* , если *conf%* транзакций из M, содержащих A, также содержат B.

Пользуясь данными из табл. 1 и формулами (1) и (2) рассчитаем поддержку и достоверность АП, когда А = «альбом», В = «фото»:

Поддержка: *supp(альбом→фото) = supp(альбом ∪ фото)* = 4/14*100=28,6%

Достоверность: *conf(альбом→фото) = supp(альбом ∪ фото) / supp(альбом) =* = 4/4*100=100%

Таким образом, ассоциация «альбом→фото» имеет невысокую поддержку, но ее достоверность равна 100%, поэтому данная ассоциация может рассматриваться как АП.

Цель анализа связей – получить возможные АП вида $A \rightarrow B$ для всех элементов с различными значениями поддержки и достоверности, которые должны быть выше определенных порогов, называемых соответственно минимальной поддержкой (*min-*

support) и минимальной достоверностью (*minconfidence*). Величины порогов зависят от конкретной задачи и, как правило, выбираются экспериментально.

Методика применения анализа связей к БПТ состоит из двух этапов:

1 Поиск всех наборов элементов, поддержка которых больше либо равна *minsupport*. Такие наборы элементов называются *популярными наборами* (*frequent itemset*).

2 Разработка АП на основе популярных наборов с достоверностью большей либо равной *minconf*.

На этапе *поиска популярных наборов* можно выделить две процедуры: генерация наборов и расчет поддержки набора. Первые алгоритмы поиска популярных наборов (AIS и SETM) генерировали наборы и рассчитывали поддержку во время чтения транзакций из базы данных, не используя при этом свойство антимонотонности, заключающееся в том, что поддержка любого набора элементов не может превышать минимальной поддержки любого из его поднаборов.

Сокращение времени поиска популярных наборов можно добиться за счет использования алгорима Apriori. Работа данного алгоритма состоит из некоторого числа (проходов) повторяющихся процедур генерации *k*-элементных наборов-кандидатов (*candidate generation*) и подсчета поддержки наборов-кандидатов (*candidate counting*). При этом процедура генерация кандидатов, заключающаяся в создании k-элементных множеств (*k* – номер этапа) выполняется во время чтения транзакций из базы данных. Далее запускается отдельная процедура вычисления поддержки каждого *k*-элементного множества, после выполнения которой происходит удаление наборов-кандидатов, поддержка которых меньше *minsupport*. Оставшиеся *k*-элементные наборы считаются популярными (*frequent itemset*) [6].

Пользуясь данными БПТ (табл. 1) покажем процесс поиска популярных наборов (*f_itemset*) на основе алгоритма Apriori, который начинается с генерации 1-элементных множеств кандидатов i_k, $k = 1 \div n$, где n – количество всех возможных элементов в базе данных. Для данных из БПТ – $n = 9$. Результат выполнения процедуры генерации представлен в виде нормализованной таблицы (табл. 2), каждая строка которой содержит транзакцию с номером TID, представляющую собой бинарный вектор T. Элемент вектора $t[k] = 1$, если i_k элемент присутствует в транзакции, иначе $t[k] = 0$.

Таблица 2 БПТ 1-элементных наборов в нормализованном виде

Эл-ты i_k	i_1	$\mathbf{i_2}$	$\mathbf{i_3}$	$\mathbf{i_4}$	$\mathbf{i_5}$	i_6	i_7	$\mathbf{i_8}$	i_9
TID	флешки	**онлайн**	**фото**	**альбом**	**рамки**	купить	печать	**магазин**	интернет
1	1	**1**	**0**	**0**	**0**	0	0	**0**	0
2	0	**0**	**1**	**1**	**0**	0	0	**0**	0
3	0	**1**	**1**	**1**	**0**	0	0	**0**	0
4	0	**0**	**1**	**0**	**1**	0	0	**0**	0
5	0	**1**	**1**	**0**	**1**	1	0	**0**	0
6	0	**1**	**1**	**0**	**0**	0	1	**0**	0
7	0	**0**	**1**	**0**	**0**	0	0	**1**	0
8	0	**1**	**1**	**0**	**1**	0	0	**0**	0
9	0	**0**	**1**	**0**	**0**	0	1	**0**	0
10	0	**0**	**1**	**0**	**0**	0	0	**1**	1
11	0	**0**	**1**	**1**	**0**	0	0	**1**	0
12	0	**0**	**1**	**0**	**1**	0	0	**1**	0
13	0	**0**	**1**	**1**	**1**	0	0	**0**	0
14	1	**0**	**0**	**0**	**0**	0	1	**0**	0
$Supp^1(i_k)$	14,3 %	**35,7%**	**85,7%**	**28,6%**	**35,7%**	7,14%	21,43%	**28,6%**	7,14%

Поддержка $supp^1(i_к)$ каждого 1-элементного набора $i_к$ вычисляется как: $\sup p^1(i_k) = (\sum_{j=1}^{m} i_{kj} / m) * 100$, где m – количество транзакций (m = 14). При этом популярным считается набор, для которого выполняется условие $supp^1(i_к)$ >= *minsupp*. Допустим, что экспериментально выбранный порог поддержки составил *minsupp* = 28,6%, тогда согласно свойству антимонотонности поддержки наборов элементов, в генерации всех возможных 2-элементных наборов-кандидатов вместо исходных девяти 1-элементных множеств будут участвовать только пять признанных популярными – $\{i_2\}$,$\{i_3\}$,$\{i_4\}$, $\{i_5\}$ и $\{i_8\}$ (табл. 2). Результаты первого, второго и третьего прохода процедур поиска популярных наборов согласно алгоритму Apriori показаны в виде решетки связей начинающейся с пустого набора (рис. 2). При этом на 1 уровне располагаются 1-элементные наборы, на 2-м – 2-элементные и т.д. Тогда на k-м уровне могут быть представлены k-элементные наборы, связанные со всеми своими (k-1)-элементными поднаборами. При этом если 1-элементный набор $\{i_1\}$ согласно (1) имеет поддержку ниже заданного порога и, соответственно, не является популярным, тогда, согласно свойству антимонотонности, все его постнаборы также не являются популярными и отбрасываются. Далее 1-элементный набор $\{i_1\}$ не участвует в генерации 2- и 3-элементных наборов. Таким образом, любой k-элементный набор будет популярным тогда и только тогда, когда все его (k-1)-элементные поднаборы будут популярными

Таким образом, использование алгоритма Apriori для поиска популярных наборов позволит резко снизить количество операций перебора всех возможных наборов элементов из БПТ, которое с ростом числа элементов в I ($|I|$) растет экспоненциально.

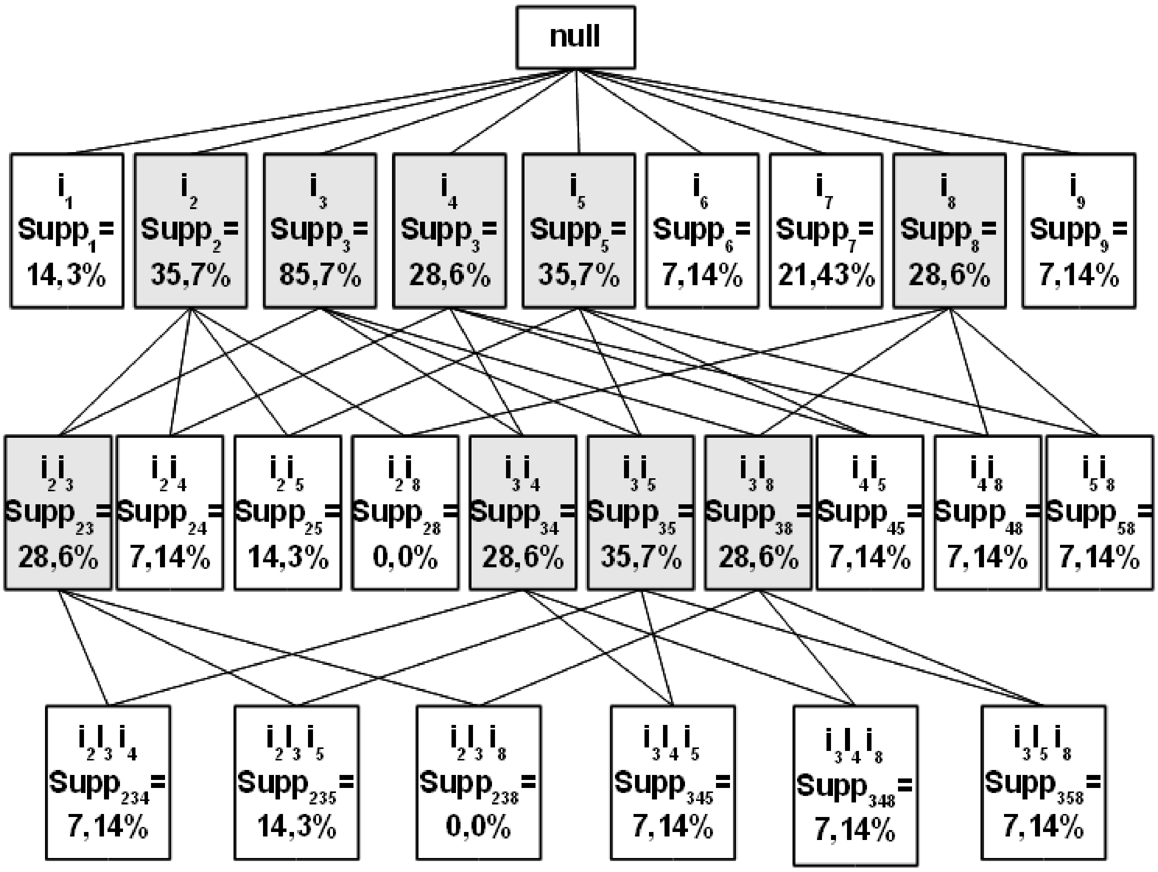


Рисунок 2 – Результаты работы первого, второго и третьего прохода процедуры поиска популярных наборов в БПТ

Количество повторяющихся проходов процедуры поиска популярных наборов согласно алгоритму Apriori меньше или равно количеству элементов в самом длинном наборе – $L = max\ (D_{TID})$, где *TID* – номер транзакции. В рассматриваемом примере (табл. 1) самым длинным является 4-элементный набор для транзакции с пятым номером, т.е $L = 4$. Однако поддержка ни одного из 3-эмементных наборов-кандидатов не больше *minsupp* (рис. 2), поэтому процесс поиска популярных наборов завершается уже после 3-го прохода, и можно переходить ко второму этапу анализа связей в БПТ – разработки АП на основе найденных популярных наборов.

Разработка АП для найденных популярных наборов *f_itemset* выполняется на основе расчета поддержки и достоверности, используя (1) и (2) для всех импликаций типа $A \rightarrow B$. При этом в качестве *A* используются все возможные популярные и непустые (*k*-1)-элементные поднаборы $f_itemset_{k-1}$ популярного k-элементного набора $f_itemset_k$. *A* в качестве *B* используется разности *R* между $f_itemset_k$ и всеми $f_itemset_{k-1}$. Например (рис.2), для набора $f_itemset_3=\{i_2i_3i_5\}$, поддержка котрого $supp^3(i_2i_3i_5)$=14,3%, поднаборами будут $f_itemset_2=\{\{i_2i_3\}, \{i_2i_5\}, \{i_3i_5\}\}$, а разностями $R = \{\{i_5\}, \{i_3\}, \{i_2\}\}$соответственно. Тогда все импликации $A \rightarrow B$ будут выглядеть, как $A \rightarrow B = \{(\{i_2i_3\} \rightarrow \{i_5\}),(\{i_2i_5\} \rightarrow \{i_3\}), (\{i_3i_5\} \rightarrow \{i_2\})\}$. При этом импликация $A \rightarrow B$ будет относится к АП тогда и только тогда, когда $supp(A \rightarrow B) > minsupp$ и $conf(A \rightarrow B) > minconf$.

При условии, что для данных из БПТ (табл.1) экспериментально выбранный порог поддержки составил *minsupp* = 28,6%., то для разработки АП будут использованы только 2-элементные популярные наборы $f_itemset_2$, т.к. поддержка $supp^3(\ldots)$ всех 3-эмементных наборов-кандидатов меньше *minsupp* (рис. 2).

Сформируем все возможные импликации типа $A \rightarrow B$ для 2-элементных популярных наборов $f_itemset_2 = \{\{i_2i_3\}, \{i_3i_4\}, \{i_3i_5\}, \{i_3\boldsymbol{i_8}\}\}$. Для этого определим их 1-элементные популярные поднаборы $f_itemset_1$ и разности *R*. Рассчитаем их поддержку и достоверность, результаты запишем в табл. 3. При условии, что экспериментально выбранный порог достоверности составит *minconf* = 60%, к АП можно отнести следующие правила: «Если фото то онлайн»; «Если фото то альбом»; «Если фото то рамки»; «Если фото то магазин».

Таблица 3 – Наборы-кандидаты в АП типа импликации $A \rightarrow B$

$f_itemset_2$	$f_itemset_1$	*R*	$A \rightarrow B$	$supp(A \rightarrow B)$	$conf(A \rightarrow B)$
$\{i_2\ i_3\}$ онлайн фото	$\{i_2\}$*онлайн*	$\{i_3\}$*фото*	*онлайн* → *фото*	(4)28,6%	(4/12) 33%
	$\{i_3\}$ *фото*	$\{i_2\}$*онлайн*	***Фото →онлайн***	**(4)28,6%**	**(4/5) 80%**
$\{i_3\ i_4\}$ фото альбом	$\{i_3\}$ *фото*	$\{i_4\}$ *альбом*	***Фото → альбом***	**(4)28,6%**	**(4/4) 100%**
	$\{i_4\}$ *альбом*	$\{i_3\}$ *фото*	*Альбом → фото*	(4)28,6%	(4/12) 33%
$\{i_3\ i_5\}$ фото рамки	$\{i_3\}$ *фото*	$\{i_5\}$ *рамки*	***фото → рамки***	**(5)35,7%**	**(5/5) 100%**
	$\{i_5\}$ *рамки*	$\{i_3\}$ *фото*	*рамки →фото*	(5)35,7%	(5/12)41,6%
$\{i_3\ i_8\}$ фото магазин	$\{i_3\}$ *фото*	$\{i_8\}$ *магазин*	***фото →магазин***	**(4)28,6%**	**(4/4) 100%**
	$\{i_8\}$ *магазин*	$\{i_3\}$ *фото*	*Магазин → фото*	(4)28,6%	(4/12) 33%

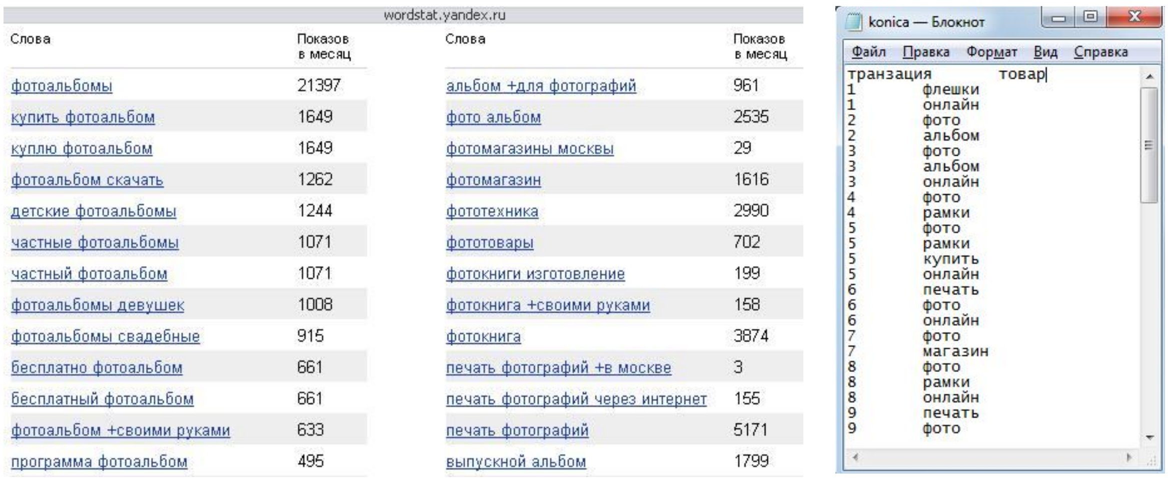
Методика автоматизации разработки и обновления СЯС

В результате анализа этапов и процедур разработки СЯС (рис. 1 б) и возможностей применения анализа связей к базе поисковых транзакций, методику автоматизации разработки и обновления СЯС можно представить в виде последовательности следующих шагов:

1 Оценка контента сайта и исследование поисковых тенденций для определения первичного списка поисковых транзакций. Например, для интернет-витрины Konica-Digital шаблон поисковой транзакции согласно территориального расположения будет выглядеть как «фото онлайн в Одессе», а согласно наивысшей стоимости – «фотографии печать».

2 С помощью средств статистики поисковых системы yandex (wordstat.yandex) формирование списка ассоциированных запросов с указанной частотой ключевых слов (рис. 3 а).

3 Формирование базы поисковых транзакций (БПТ) заданного вида на основании списка ассоциированных запросов (рис. 3 а)



wordstat.yandex.ru

Слова	Показов в месяц	Слова	Показов в месяц
фотоальбомы	21397	альбом +для фотографий	961
купить фотоальбом	1649	фото альбом	2535
куплю фотоальбом	1649	фотомагазины москвы	29
фотоальбом скачать	1262	фотомагазин	1616
детские фотоальбомы	1244	фототехника	2990
частные фотоальбомы	1071	фототовары	702
частный фотоальбом	1071	фотокниги изготовление	199
фотоальбомы девушек	1008	фотокнига +своими руками	158
фотоальбомы свадебные	915	фотокнига	3874
бесплатно фотоальбом	661	печать фотографий +в москве	3
бесплатный фотоальбом	661	печать фотографий через интернет	155
фотоальбом +своими руками	633	печать фотографий	5171
программа фотоальбом	495	выпускной альбом	1799

а б

Рисунок 3 – Экранные формы автоматизации разработки СЯС Konica-Digital
(а – статистики поисковых запросов в wordstat.yandex;
б – вида БПТ в текстовом формате)

4 Поиск популярных наборов в БПТ на основе алгоритма Apriori и формирование базы популярных поисковых транзакций (БППТ).

5 На основе БППТ формирование базы возможных импликаций типа «условие → следствие», расчет их поддержки и достоверности и формирование базы ассоциативных правил.

6 Формирование META-тэгов (Title, Description, Keywords) и, возможно, модификация контента сайта. Например, для интернет-витрины Konica-Digital мета тег keywords, будет выглядеть так: *<meta name="keywords" content="интернет магазин, магазин интернет, купить онлайн, фотографии онлайн, фотографии печать, фотографии рамки, магазин рамки интернет, купить онлайн рамки, ...">*

Выводы

Реализация предлагаемой методики разработки СЯС с динамическим контентом позволила поднять позиции Konica-Digital в SERP на 25% для 70% информационных, 85% транзакционных и 60% нечетких запросов, вводимых пользователем в основные ПС Yandex и Google. При этом в 1,5 раза сократились затраты рабочего времени специалиста по SEO, необходимые для достижения заявленных результатов.

Ограниченный объем статьи не позволил показать другие приложения методики разработки СЯС с динамическим контентом. Однако необходимо заметить, что при реализации предлагаемой методики для интернет-магазина Vsedetali в качестве БПТ использовалась таблица заказов, а автоматизированное формирование атрибута content мета тэгов keywords на основе АП также позволило повысить полноту и точность, снизить время разработки семантического ядра сайта. Таким образом, предложенная методика разработки СЯС является достаточно универсальной и с небольшими доработками может быть применена специалистами по SEO для эффективного продвижения сайтов с динамическим контентом

Литература

1. Ашманов И. Оптимизация и продвижение сайтов в поисковых системах / И. Ашманов, А. Иванов. – [3-е изд.]. – СПб. : Питер, 2011. – 464 с.
2. Как работают поисковые системы – сниппет, алгоритм обратных индексов, индексация страниц, особенности работы поисковиков [Электронный ресурс]. – Режим доступа : http://ktonanovenkogo.ru/seo/search/kak-rabotayut-poiskovye-sistemy-snippet-index.html – 2009.
3. Паклин Н.Б. Бизнес-аналитика: от данных к знаниям / Н.Б. Паклин, В.И. Орешков. – СПб. : Изд. Питер, 2009. – 624 с.
4. Dirk Chung. Suchmaschienen-Optimierung: Darschnell Einstieg / Dirk Chung, Andreas Klünder. – Heidelberg, 2007.
5. Timo Aden. Google Analytics: 2 Auflage / Timo Aden. – München : Carl Hanser Verlag, 2010.
6. Agrawal R. 1993. Mining Associations between Sets of Items in Massive Databases / R. Agrawal, T. Imielinski, A. Swami. // Proc. of the ACM-SIGMOD Int'l Conf. on Management of Data, 1993. – P. 207-216

Literatura

1. Ashmanov I. Optimizacija i prodvizhenie sajtov v poiskovyh sistemah: 3-e izd. SPb.: Piter. 2011. 464 s.
2. Kak rabotajut poiskovye sistemy – snippet, algoritm obratnyh indeksov, indeksacija stranic, osobennosti raboty poiskovikov. http://ktonanovenkogo.ru/seo/search/kak-rabotayut-poiskovye-sistemy-snippet-index.html – 2009 g.
3. Paklin N.B. . Biznes-analitika: ot dannyh k znanijam. SPb.: Izd. Piter. 2009. 624 s.
4. Dirk Chung. Suchmaschienen-Optimierung: Darschnell Einstieg. REDLINE GmbH. Heidelberg. 2007.
5. Timo Aden. Google Analytics: 2 Auflage. Carl Hanser Verlag München. 2010.
6. Agrawal R.. In Proc. of the 1993 ACM-SIGMOD Int'l Conf. on Management of Data. 1993. S. 207-216

Стаття поступила в редакцию 01.06.2012.